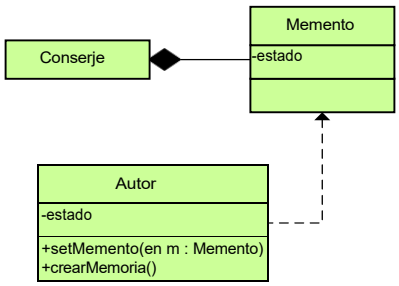
C Fábrica abstracta
S Adaptador
S Puente
C Constructor
B Cadena de responsabilidad
B Comando
S Compuesto
S Decorador
S Fachada
C Método de fábrica
S Peso mosca
B Intérprete
B Iterador
B Mediador
B Memento
C Prototipo
S Proxy
B Observador
C Singleton
B Estado
B Estrategia
B Método de plantillas
B Visitante

Memento

Tipo: Comportamiento

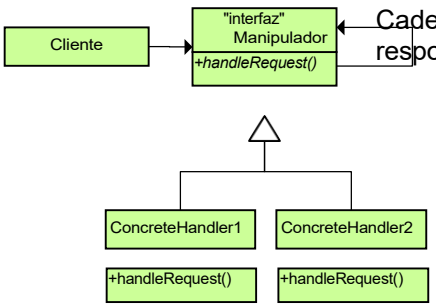
Lo que es:
Sin violar la encapsulación, capturar y externalizar el estado interno de un objeto para que el objeto pueda ser restaurado a este estado más tarde.



Cadena sucesoria de responsabilidad

Tipo: Comportamiento

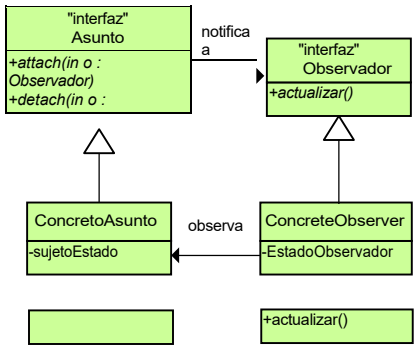
Lo que es:
Evita acoplar el emisor de una petición a su receptor dando a más de un objeto la oportunidad de gestionar la petición. Encadena los objetos receptores y pasa la solicitud a lo largo de la cadena hasta que un objeto la gestione.



Observador

Tipo: Comportamiento

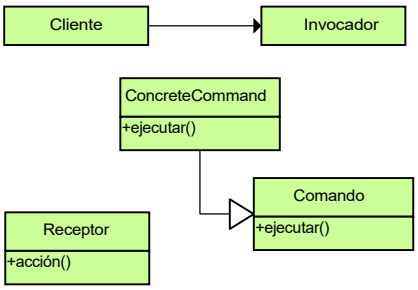
Lo que es:
Defina una dependencia de uno a muchos entre objetos para que cuando un objeto cambie de estado, todos sus dependientes sean notificados y actualizados automáticamente.



Comando

Tipo: Comportamiento

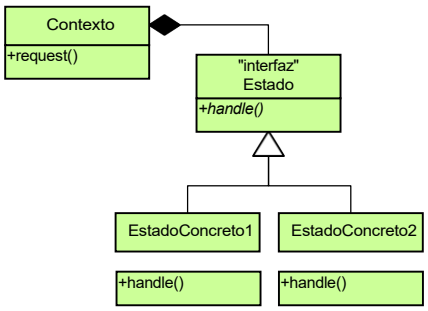
Lo que es:
Encapsular una solicitud como un objeto, lo que le permite parametrizar clientes con diferentes solicitudes, poner en cola o registrar solicitudes y admitir operaciones que se pueden deshacer.



Estado

Tipo: Comportamiento

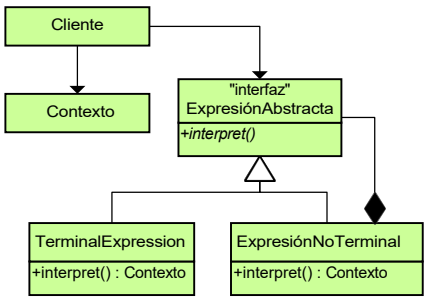
Lo que es:
Permite que un objeto altere su comportamiento cuando cambia su estado interno. El objeto parecerá cambiar de clase.



Intérprete

Tipo: Comportamiento

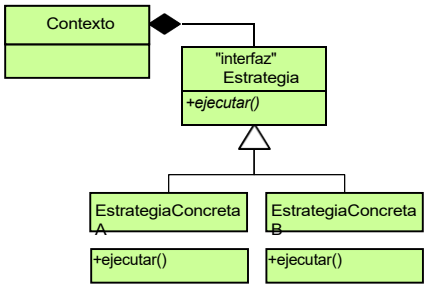
Lo que es:
Dado un lenguaje, defina una representación para su gramática junto con un intérprete que utilice la representación para interpretar sentencias en el lenguaje.



Estrategia

Tipo: Comportamiento

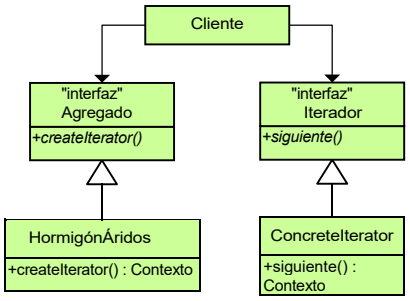
Lo que es:
Define una familia de algoritmos, encapsula cada uno de ellos y los hace intercambiables. Permite que el algoritmo varíe independientemente de clientes que lo utilizan.



Iterador

Tipo: Comportamiento

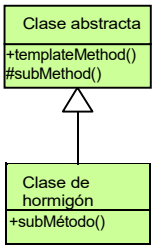
Lo que es:
Proporcionar una forma de acceder a los elementos de un objeto agregado secuencialmente sin exponer su representación subyacente.



Método de plantillas

Tipo: Comportamiento

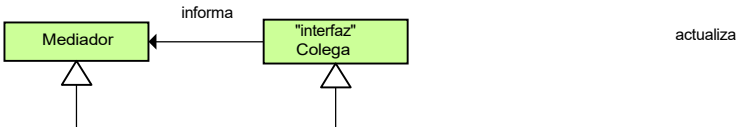
Lo que es:
Definir el esqueleto de un algoritmo en una operación, difiriendo algunos pasos a subclases. Permite a las subclases redefinir ciertos pasos de un algoritmo sin cambiar la estructura del algoritmo.



Mediador

Tipo: Comportamiento

Lo que es:
Define un objeto que encapsula cómo interactúa un conjunto de objetos. Promueve



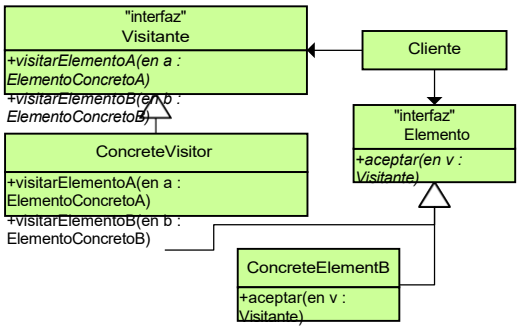
el acoplamiento flexible al evitar que los objetos se refieran entre sí explícitamente y permite variar sus interacciones de forma independiente.

Visitante

Tipo: Comportamiento

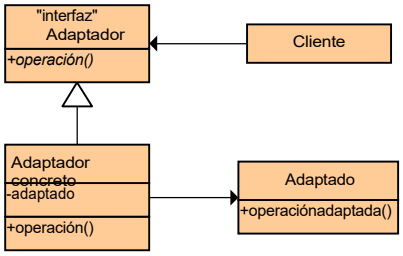
Lo que es:
Representar una operación a realizar sobre los elementos de una estructura de objetos. Permite definir

ir una nueva operación sin cambiar las clases de los elementos sobre los que opera.



Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John (1995). *Design Patterns: Elements of Reusable Object-Oriented Software*. Reading, Massachusetts: Addison Wesley Longman, Inc.

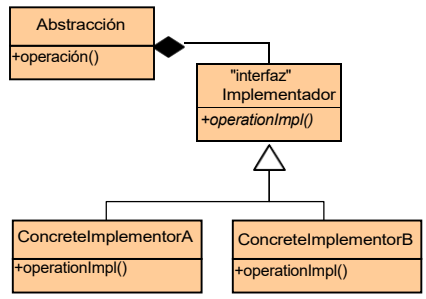
ConcreteElementA
+aceptar(en v : Visitante)



Adaptador

Tipo: Estructural

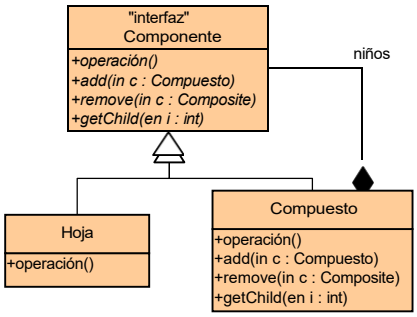
Lo que es:
Convierte la interfaz de una clase en otra interfaz que esperen los clientes. Permite que funcionen juntas clases que de otro modo no podrían debido a interfaces incompatibles.



Puente

Tipo: Estructural

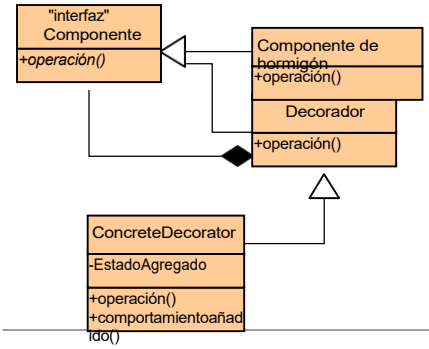
Lo que es:
Desacoplar una abstracción de su implementación para que ambas puedan variar independientemente.



Compuesto

Tipo: Estructural

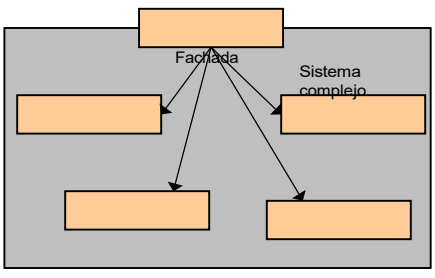
Lo que es:
Componga objetos en estructuras de árbol para representar jerarquías parte-todo. Permite a los clientes tratar objetos individuales y composiciones de objetos de manera uniforme.



Decorador

Tipo: Estructural

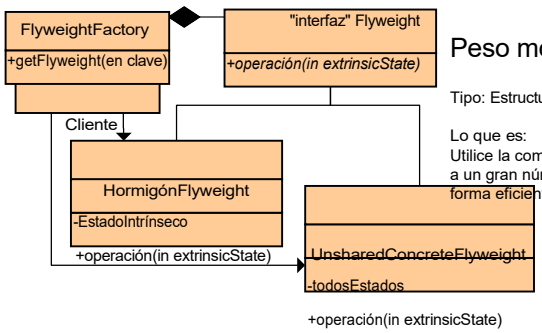
Lo que es:
Adjuntar responsabilidades adicionales a un objeto de forma dinámica. Proporcionan una alternativa flexible a las subclases para ampliar la funcionalidad.



Fachada

Tipo: Estructural

Lo que es:
Proporcionar una interfaz unificada a un conjunto de interfaces de un subsistema. Define una interfaz de alto nivel que facilita el uso del subsistema.



Peso mosca

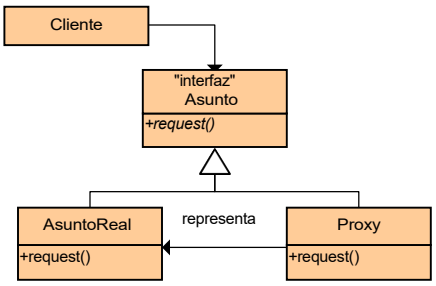
Tipo: Estructural

Lo que es:
Utilice la compartición para dar soporte a un gran número de objetos finos de forma eficiente.

Proxy

Tipo: Estructural

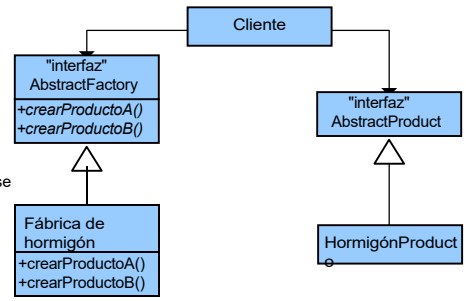
Lo que es:
Proporcionar un sustituto o marcador de posición de otro objeto para controlar el acceso al mismo.



Fábrica abstracta

Tipo: Creación

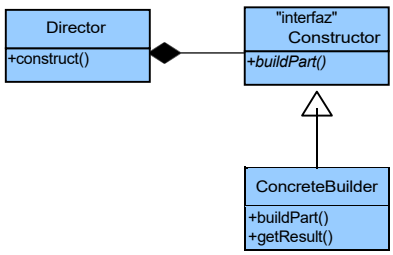
Lo que es:
Proporciona una interfaz para crear familias de objetos relacionados o dependientes sin especificar su clase concreta.



Constructor

Tipo: Creación

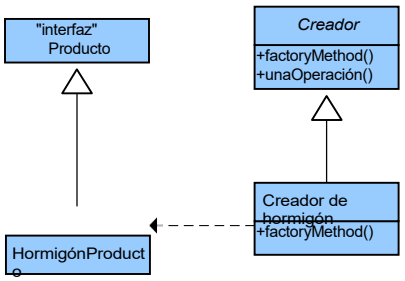
Lo que es:
Separar la construcción de un objeto complejo de su representación para que un mismo proceso de construcción pueda crear diferentes representaciones.



Método de fábrica

Tipo: Creación

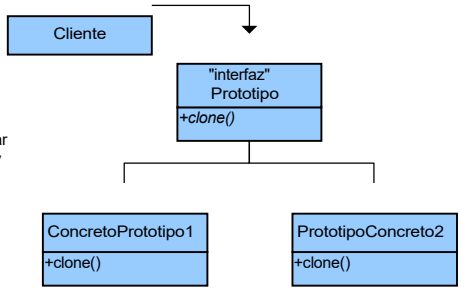
Lo que es:
Define una interfaz para crear un objeto, pero deja que las subclases decidan qué clase instanciar. Permite a una clase diferir la instanciación a las subclases.



Prototipo

Tipo: Creación

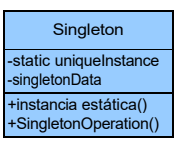
Lo que es:
Especifica los tipos de objetos a crear utilizando una instancia prototípica, y crea nuevos objetos copiando este prototipo.



Singleton

Tipo: Creación

Lo que es:
Asegúrese de que una clase sólo tiene una instancia y proporcione un punto de acceso global a la misma.



Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John (1995). *Design Patterns: Elements of Reusable Object-Oriented Software*. Reading, Massachusetts: Addison Wesley Longman, Inc.